$ifclassloadedbeamer$

1D Convolutional Neural Networks

Nipun Batra

IIT Gandhinagar

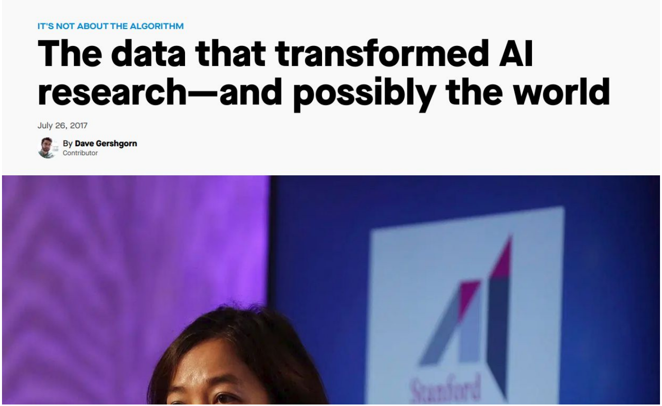
July 29, 2025

Convolutional Neural Networks

Imagenet

14 million images, 20K categories

Imagenet


IT'S NOT ABOUT THE ALGORITHM

The data that transformed AI research—and possibly the world

July 26, 2017

By **Dave Gershgorn**
Contributor

https://qz.com/1034972/the-data-that-changed-the-direction-of-ai-research-and-possibly-the-world/

Imagenet

- Circa 2006, AI community: “a better algorithm would make better decisions, regardless of the data.”
- Fei Fei Li thought: “the best algorithm wouldn’t work well if the data it learned from didn’t reflect the real world”
- “We decided we wanted to do something that was completely historically unprecedented,” Li said, referring to a small team who would initially work with her. “We’re going to map out the entire world of objects.

Imagenet

- ImageNet: published in 2009 as a research poster stuck in the corner of a Miami Beach conference center, the dataset quickly evolved into an annual competition to see which algorithms could identify objects in the dataset's images with the lowest error rate.
- "The paradigm shift of the ImageNet thinking is that while a lot of people are paying attention to models, let's pay attention to data," Li said. "Data will redefine how we think about models."

WordNet

WordNet

- In the late 1980s, Princeton psychologist George Miller started a project called WordNet, with the aim of building a hierarchical structure for the English language.
- For example, within WordNet, the word “dog” would be nested under “canine,” which would be nested under “mammal,” and so on. It was a way to organize language that relied on machine-readable logic, and amassed more than 155,000 indexed words.

Back to Imagenet

- Finding the perfect algorithm seemed distant, Li says. She saw that previous datasets didn't capture how variable the world could be—even just identifying pictures of cats is infinitely complex.
- If you only saw five pictures of cats, you'd only have five camera angles, lighting conditions, and maybe variety of cat. But if you've seen 500 pictures of cats, there are many more examples to draw commonalities from.
- Having read about WordNet's approach, Li met with professor Christiane Fellbaum, a researcher influential in the continued work on WordNet, during a 2006 visit to Princeton. Fellbaum had the idea that WordNet could have an image associated with each of the words, more as a reference rather than a computer vision dataset.

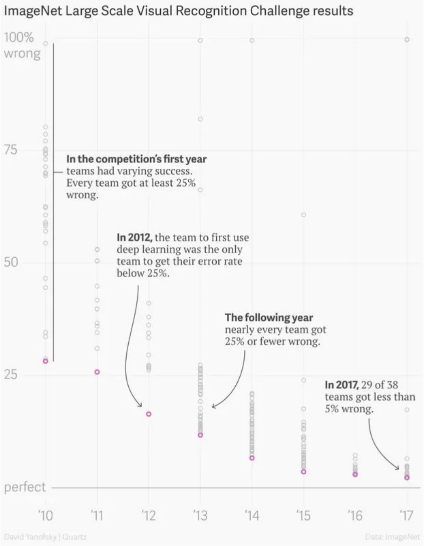
Back to Imagenet

- Li's first idea was to hire undergraduate students for $10 an hour to manually find images and add them to the dataset. But back-of-the-napkin math quickly made Li realize that at the undergrads' rate of collecting images it would take 90 years to complete.
- Undergrads were time-consuming, algorithms were flawed, and the team didn't have money—Li said the project failed to win any of the federal grants she applied for, receiving comments on proposals that it was shameful Princeton would research this topic, and that the only strength of proposal was that Li was a woman.
- A solution finally surfaced in a chance hallway conversation with a graduate student who asked Li whether she had heard of Amazon Mechanical Turk, a service where hordes of humans sitting at computers around the world would complete small online tasks for pennies.

Back to Imagenet

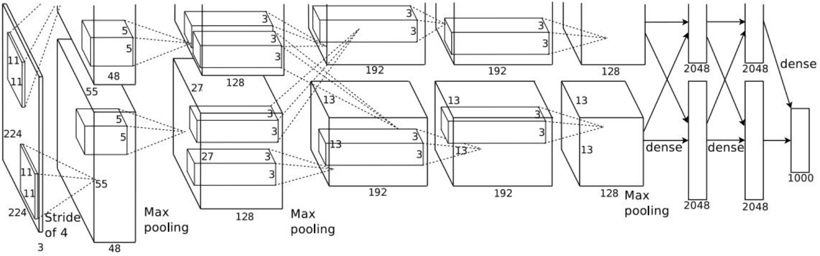
Back to Imagenet

- Even after finding Mechanical Turk, the dataset took two and a half years to complete. It consisted of 3.2 million labelled images, separated into 5,247 categories, sorted into 12 subtrees like "mammal," "vehicle," and "furniture."
- In 2009, Li and her team published the ImageNet paper with the dataset—to little fanfare. Li recalls that CVPR, a leading conference in computer vision research, only allowed a poster, instead of an oral presentation, and the team handed out ImageNet-branded pens to drum up interest. People were skeptical of the basic idea that more data would help them develop better algorithms.
- "There were comments like 'If you can't even do one object well, why would you do thousands, or tens of thousands of objects?"

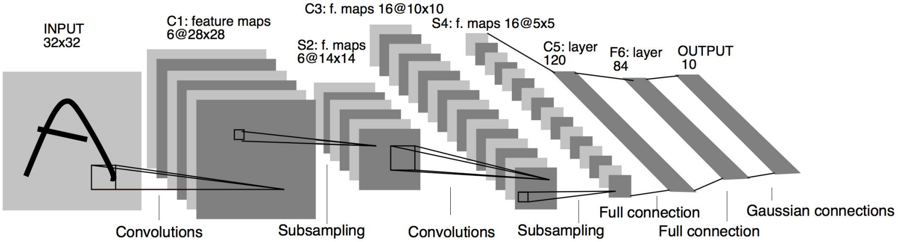


14 million images, 20K categories

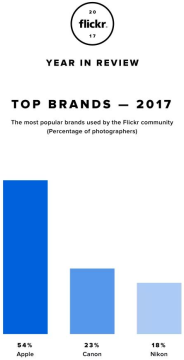
History (AlexNet 2012)



History (LeCun 1998)

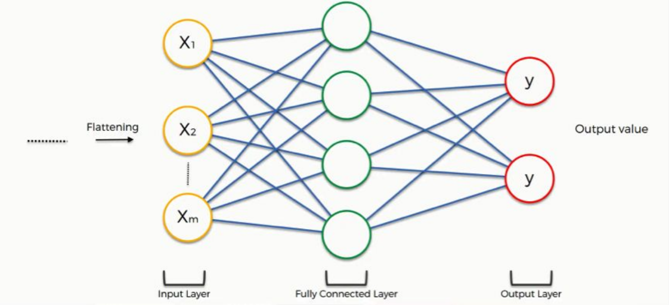


Modern day cameras



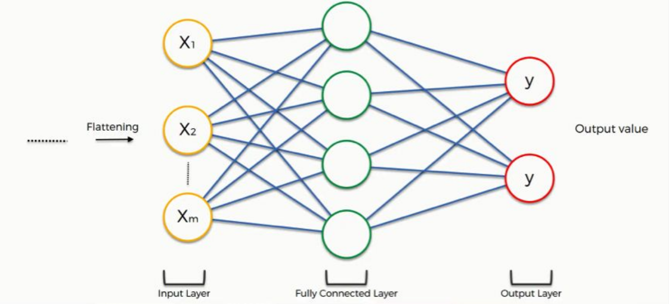
Modern day cameras

Modern day cameras suitability for MLPs?



Courtesy:
https://www.superdatascience.com/convolutional-neural-networks-cnn-step-4-full-connection/

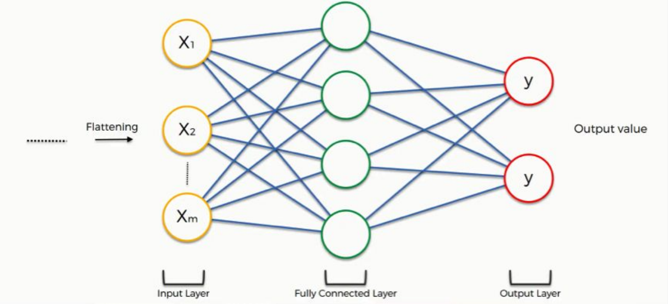
Modern day cameras suitability for MLPs?



Courtesy:
https://www.superdatascience.com/convolutional-neural-networks-cnn-step-4-full-connection/

1. If we are classifying cats vs dogs and hidden layer size is 100, what is number of parameters?

Modern day cameras suitability for MLPs?



Courtesy:
https://www.superdatascience.com/convolutional-neural-networks-cnn-step-4-full-connection/

1. If we are classifying cats vs dogs and hidden layer size is 100, what is number of parameters?
2. N[1] = 100, N[0] = 108*1M*3 (for RGB channel) → Billions of params
3. Size of weight matrix assuming each param is 32 bytes is 32 bytes*324 billion → several GBs

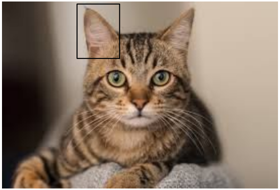
Are MLPs well suited for images?

Courtesy: https://www.rd.com/advice/pets/common-cat-myths/

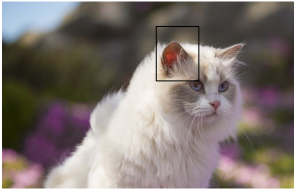
Courtesy: https://www.goodhousekeeping.com/life/pets/g21525625/why-cats-are-best-pets/

Are both of the above cats?

Are MLPs well suited for images?



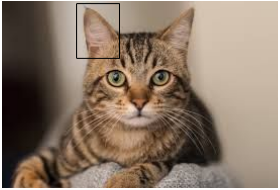
Courtesy:
https://www.rd.com/advice/pets/common-cat-myths/



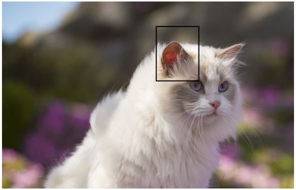
Courtesy:
https://www.goodhousekeeping.com/life/pets/g21525625/why-cats-are-best-pets/

Assume both are 100X100 images and bounded rectangle are 10X10 pixels

Are MLPs well suited for images?



Courtesy:
https://www.rd.com/advice/pets/common-cat-myths/



Courtesy:
https://www.goodhousekeeping.com/life/pets/g21525625/why-cats-are-best-pets/

A cat ear is a cat ear, irrespective of the location in the image.

MLP would see these are different input features

Rather, we need “feature detector” that is **translation invariant**.

Are MLPs well suited for images?

Similar
pixel
values

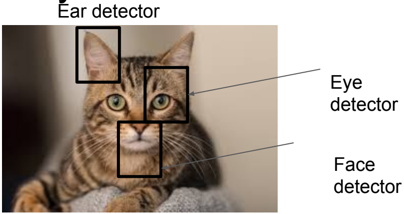
Courtesy:
https://www.rd.com/advice/pets/common-cat-myths/

Courtesy:
https://www.goodhousekeeping.com/life/pets/g21525625/why-cats-are-best-pets/

MLPs assume all input features to be independent

But, we have a **spatially local** structure, nearby pixels are similar

Key Idea



Courtesy: https://www.rd.com/advice/pets/common-cat-myths/

Courtesy: https://www.goodhousekeeping.com/life/pets/g21525625/why-cats-are-best-pets/

Build **local feature** detectors

Building Block: Filters and Convolution Operation

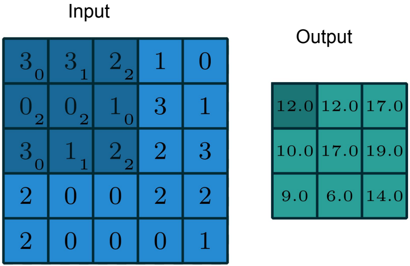
(A guide to convolution arithmetic for deep learning)

Filter

0	1	2
2	2	0
0	1	2

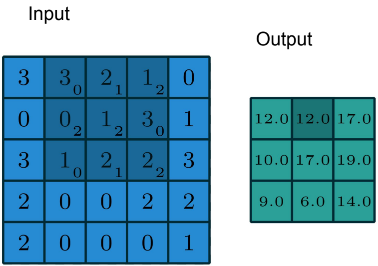
Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)



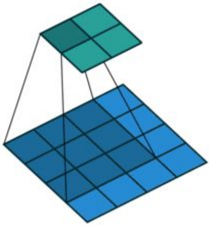
Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)



Building Block: Filters and Convolution Operation

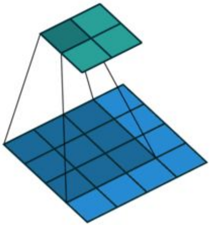
(A guide to convolution arithmetic for deep learning)



Notebook demonstration (edge detection)

Building Block: Filters and Convolution Operation

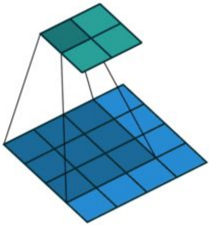
(A guide to convolution arithmetic for deep learning)



Given input image of n X n and filter of size: f X f, what is the size of the output?

Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

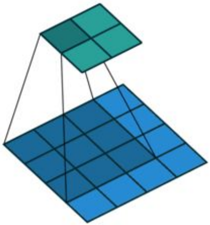


Given input image of n X n and filter of size: f X f, what is the size of the output?

n-f+1 X n-f+1

Building Block: Filters and Convolution Operation

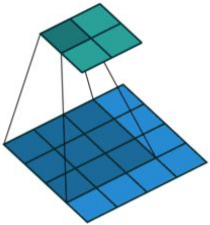
(A guide to convolution arithmetic for deep learning)



Start with a 32 X 32 image and repeated operations of a single 5 X 5 filter, after how many such operations will we have a 1 X 1 output?

Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

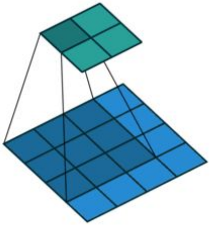


Start with a 32 X 32 image and repeated operations of a single 5 X 5 filter, after how many such operations will we have a 1 X 1 output?

Iteration	n	f	n-f+1
1	32	5	28
2	28	5	24
3	24	5	20
4	20	5	16
...	...	...	...

Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)



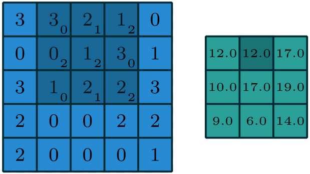
Start with a 32 X 32 image and repeated operations of a single 5 X 5 filter, after how many such operations will we have a 1 X 1 output?

Iteration	n	f	n-f+1
1	32	5	28
2	28	5	24
3	24	5	20
4	20	5	16
...	...	...	...

Problem 1: Can not go very deep with repeated convolution as image size reduces quickly

Building Block: Filters and Convolution Operation

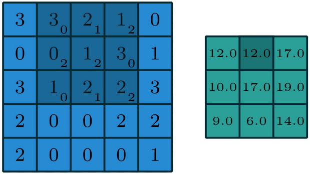
(A guide to convolution arithmetic for deep learning)



How many times is left-most pixel used in a calculation?

Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)

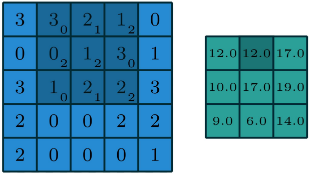


How many times is left-most pixel used in a calculation?

Only once!

Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)



How many times is left-most pixel used in a calculation?

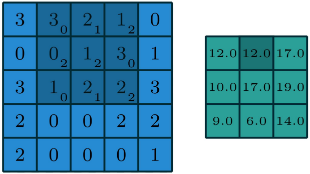
Only once!

How many times is a middle pixel used in a calculation?

Many times. For example, the middle pixel with value 2 used nine times!

Building Block: Filters and Convolution Operation

(A guide to convolution arithmetic for deep learning)



Problem 2: The corner pixels are under-utilised

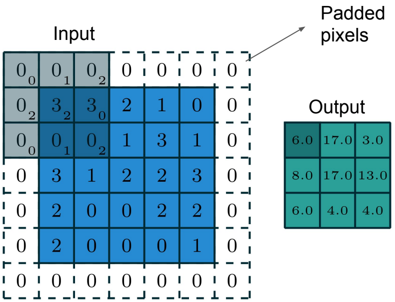
How many times is left-most pixel used in a calculation?

Only once!

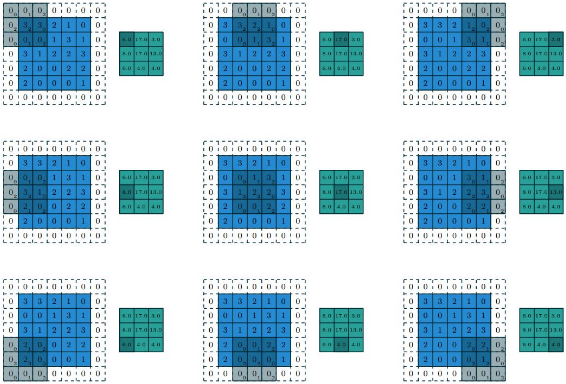
How many times is a middle pixel used in a calculation?

Many times. For example, the middle pixel with value 2 used nine times!

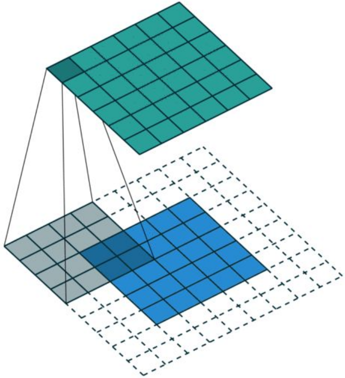
Building Block: Padding



Building Block: Padding

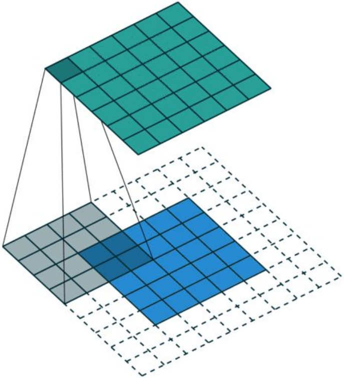


Building Block: Padding



Ques: Given padding of **p** pixel, **n X n** image and filter **f x f**, what is the output size?

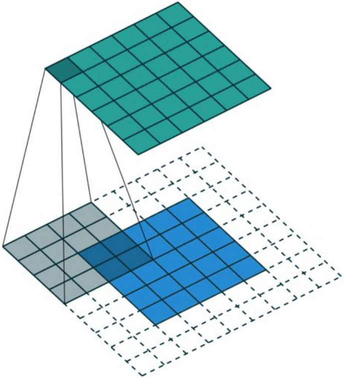
Building Block: Padding



Ques: Given padding of **p** pixel, **n X n** image and filter **f x f**, what is the output size?

n+2p-f+1 X n+2p-f+1

Building Block: Padding

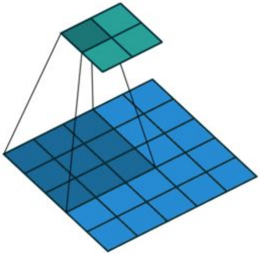


Ques: Given padding of **p** pixel, **n X n** image and filter **f x f**, what is the output size?

$n+2p-f+1 \text{ X } n+2p-f+1$

Same padding: when $n+2p-f+1 = n$ or, $p = (f-1)/2$

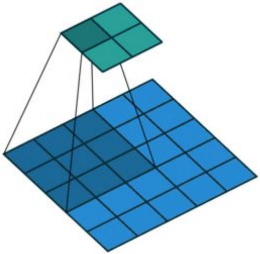
Building Block: Strides (subsampling)



Skip every **s** pixels

Ques: Given p padding, n x n image, f x f filter, s stride, what is output length?

Building Block: Strides (subsampling)

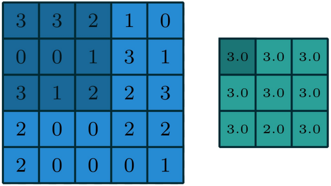


Skip every **s** pixels

Ques: Given p padding, n x n image, f x f filter, s stride, what is output length?

$\lfloor (n+2p-f)/s \rfloor + 1 \times \lfloor (n+2p-f)/s \rfloor + 1$

Building Block: Pooling (subsampling)



Max pooling

Similar to filter and convolution operation, but, gives the max value in the f x f as the output

Building Block: Pooling (subsampling)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

3.0	3.0	3.0
3.0	3.0	3.0
3.0	2.0	3.0

Max pooling

Similar to filter and convolution operation, but, gives the max value in the f x f as the output

Works well in practice
Reduces representation size

Building Block: Pooling (subsampling)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

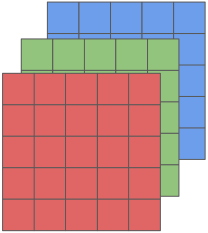
1.7	1.7	1.7
1.0	1.2	1.8
1.1	0.8	1.3

Average pooling

Similar to filter and convolution operation, but, gives the average value in the f x f as the output

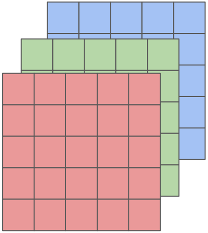
Works well in practice
Reduces representation size

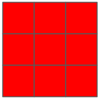
Building Block: Multiple channels



Input: n x n x c
image

Building Block: Multiple channels

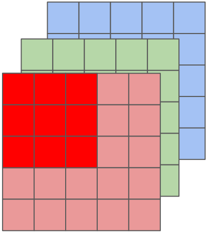




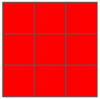
Input: n x n x c image

Filter for r channel: f x f

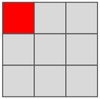
Building Block: Multiple channels



Input: n x n x c
image

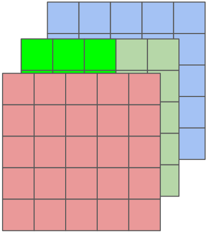


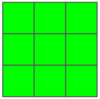
Filter for r
channel: f x f

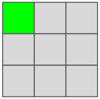


Output for r
channel: n-f+1 x
n-f+1

Building Block: Multiple channels





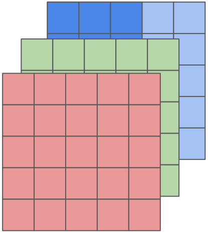


Input: n x n x c
image

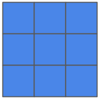
Filter for g
channel: f x f

Output for g
channel: n-f+1 x
n-f+1

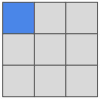
Building Block: Multiple channels



Input: n x n x c
image

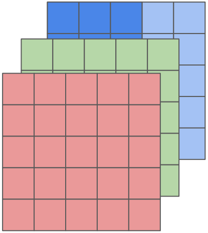


Filter for b
channel: f x f

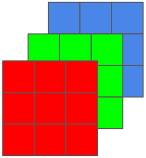


Output for b
channel: n-f+1 x
n-f+1

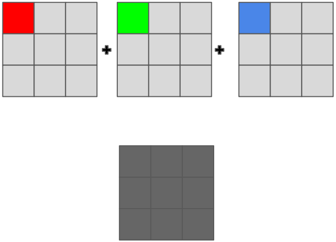
Building Block: Multiple channels



Input: n x n x c
image

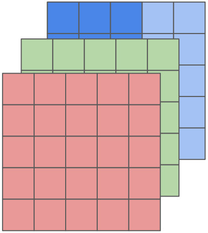


Filter for 3
channel: f x f X 3

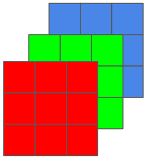


Output for 3
channel: n-f+1 x
n-f+1 X 1

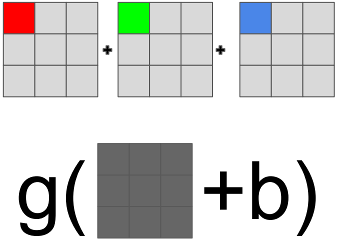
Building Block: Non-linearity



Input: n x n x c
image

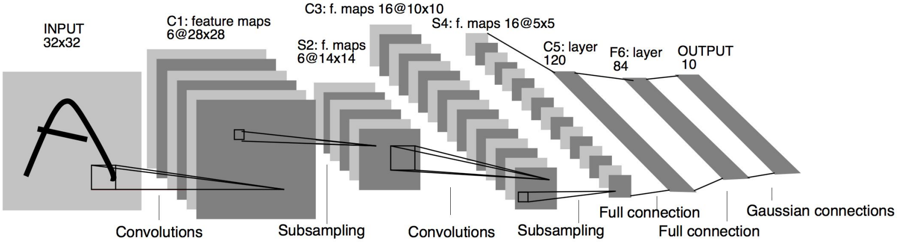


Filter for 3
channel: f x f X 3



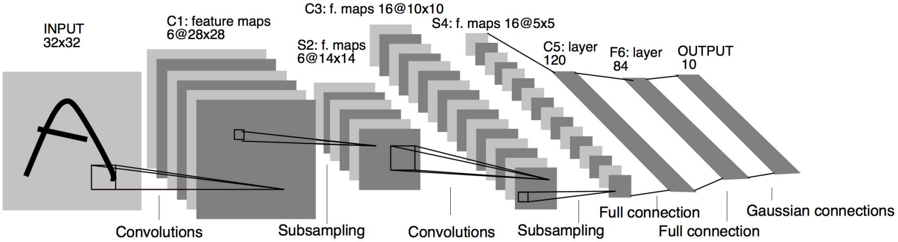
Activation Output
for 3 channel:
n-f+1 x n-f+1 X 1

Exercise LeNet-5



Exercise LeNet-5

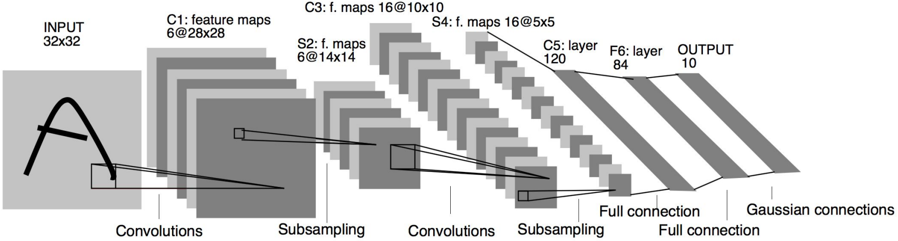
Q1: What is input size?



Exercise LeNet-5

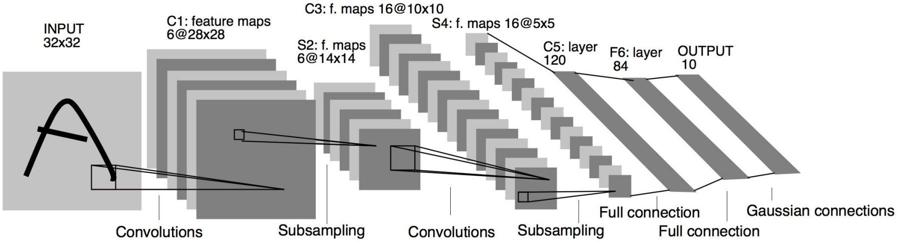
Q1: What is input size?

32X32X1 (grayscale)



Exercise LeNet-5

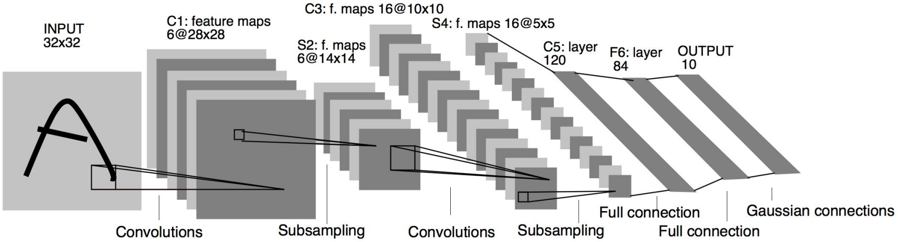
Q2: What is filter size for first layer (assume no padding)



Exercise LeNet-5

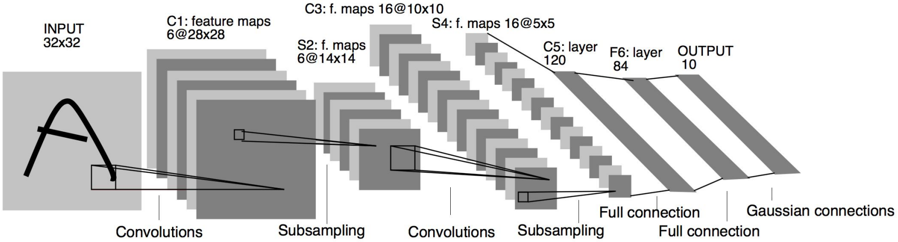
Q2: What is filter size for first layer (assume no padding, 1 stride)

5X5: 32 $\rightarrow$ 32 - 5 +1 =28



Exercise LeNet-5

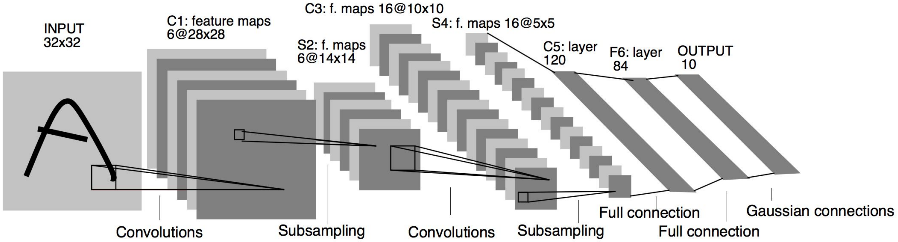
Q3: What is number of filters used in first layer?



Exercise LeNet-5

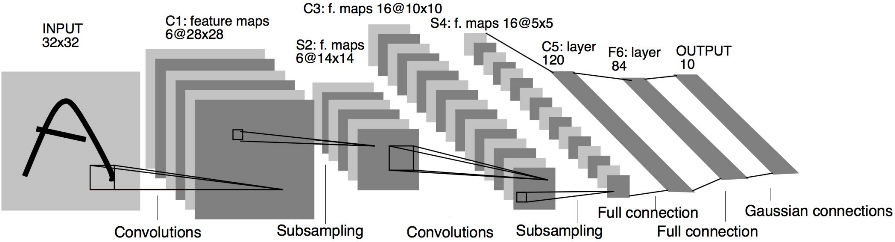
Q3: What is number of
filters used in first layer?

6



Exercise LeNet-5

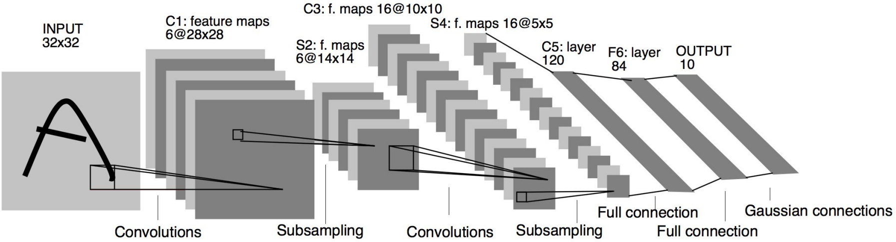
Q4: What is size of pool filter?



Exercise LeNet-5

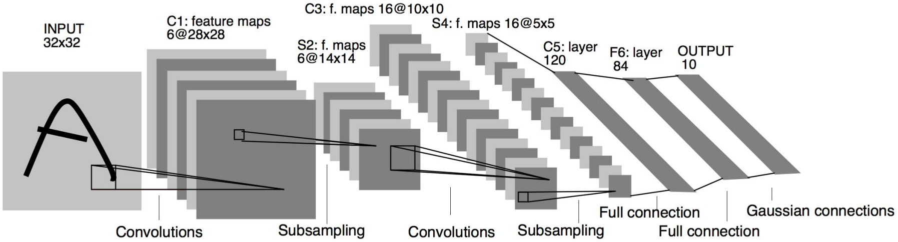
Q4: What is size of pool filter?

f=2, s=2 (stride 2)



Exercise LeNet-5

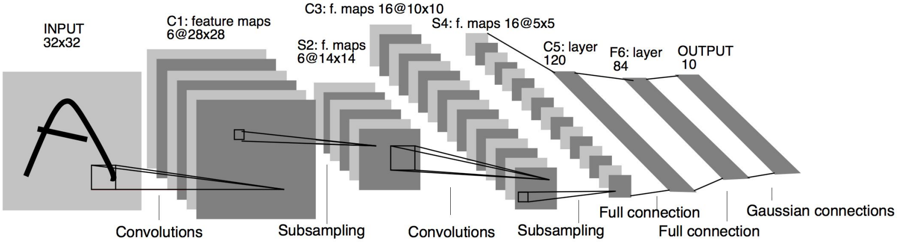
Q5: What is size of filter
for this layer convolution?



Exercise LeNet-5

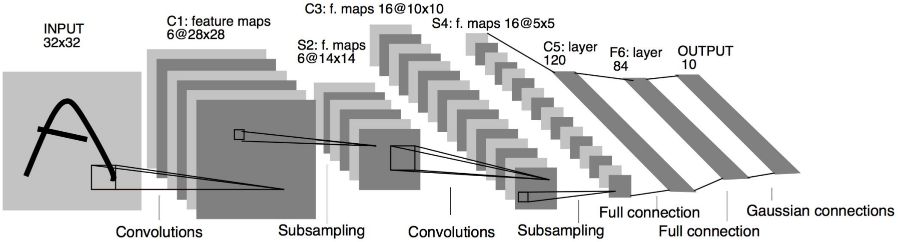
Q5: What is size and number of filter for this layer convolution?

16 filter 5X5 size with stride 1



Exercise LeNet-5

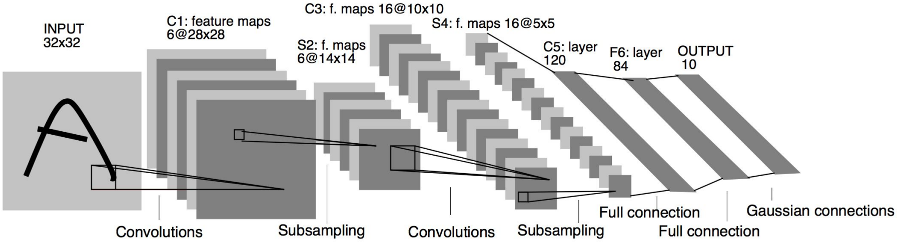
Q6: What is size of this pool layer?



Exercise LeNet-5

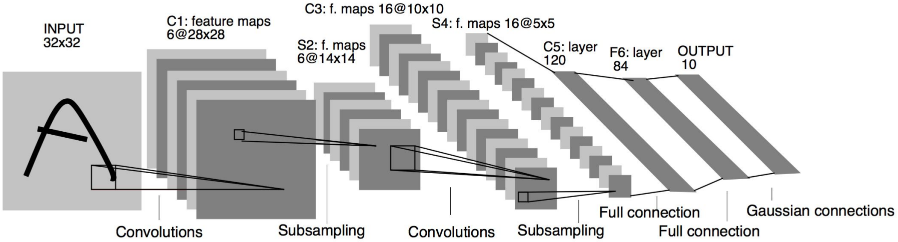
Q6: What is size of this pool layer?

f=2, s=2



Exercise LeNet-5

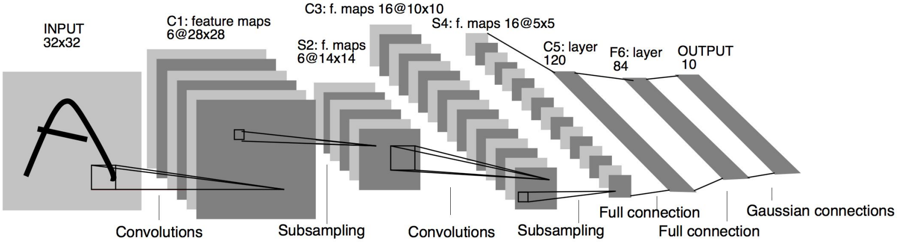
Q7: This layer is connected to an MLP like layer, how?



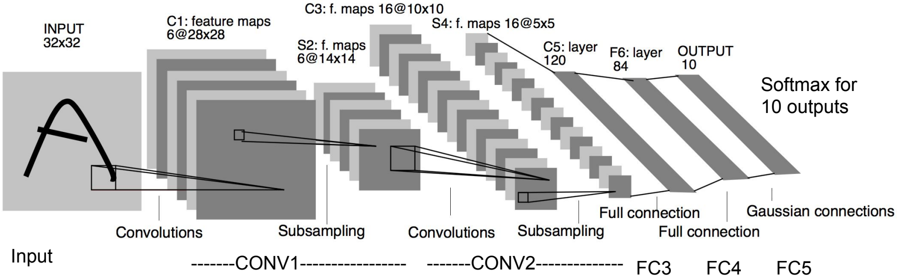
Exercise LeNet-5

Q7: This layer is connected to an MLP like layer, how?

We flatten 16X5X5 to create a 400X1 matrix

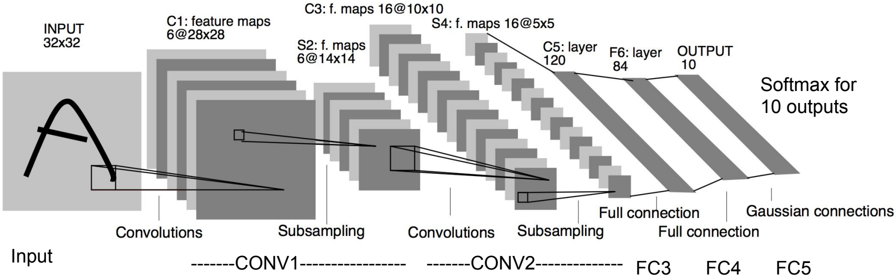


Exercise LeNet-5



Exercise LeNet-5

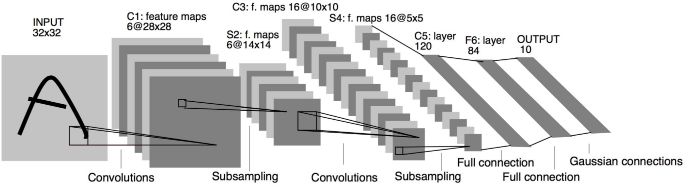
What is the total number of parameters?



Exercise LeNet-5

What is the total number of parameters?

- CONV1: 6 filters of size 5 X5X1(channel) = (6*5*5) + 6 biases = 156
- POOL1: No params
- CONV2: 16 filters of size 5 X 5X6(six channels) = (16*5*5*6) + 16 biases = 2416
- FC1: Weight matrix of size 120 X 400 + 120 biases = 48120
- FC2: Weight matrix of size 84 X 120 + 84 biases = 10164
- FC3: Weight matrix of size 10 X 84 + 10 biases = 850
- Total = 61,706



Notebook: LeNet-5, AlexNet, VGG-16

- Notebook

Training CNNs for own applications

- Train fully from scratch
- Transfer learning -- store activations

Visualising CNNs

- t-SNE or PCA on last hidden layer … MNIST
- Same exercise on Imagenet? ..